IN THE CLAIMS:

Please amend the claims as follows.

1.   (Currently amended) A dirty memory control logic for a computer system, the dirty memory being operable to store dirty indicators, each dirty indicator being settable to a given value indicative that a block of memory associated therewith has been dirtied, wherein said dirty indicators being stored in groups with each group having associated therewith a validity indicator computed from the dirty indicator values of the group, wherein the control logic being operable on reading [[a]] said group to compute a validity indicator value based on the dirty indicator values for the group to determine ~~the~~ an integrity of the group, wherein the control logic is configured to identify all dirty indicators of a group as representing a dirtied state where it computes a validity indicator value based on the dirty indicator values read for the group that is different from a validity indicator value read for that group.

2.   (Original) The dirty memory control logic of claim 1, wherein the validity indicator is a parity indicator.

3.   (Cancelled)

4.   (Currently amended) The dirty memory control logic of claim [[3]] 1, wherein if ~~operable to cause the block of memory associated with each dirty indicator of a group to be copied from the memory to another memory where~~ the control logic computes a validity indicator value based on the dirty indicator values read for the group that is different from a validity indicator value read for that group, the control logic is operable to cause the block of memory associated with each dirty indicator of the group to be copied from the memory to another memory.

5.   (Cancelled)

2

6. (Original) The dirty memory control logic of claim 1, wherein a group of dirty indicators plus the validity indicator occupy one memory word.

7. (Original) The dirty memory control logic of claim 1, wherein each dirty indicator comprises a single bit.

8. (Original) The dirty memory control logic of claim 1, wherein a validity indicator comprises a single bit.

9. (Original) The dirty memory control logic of claim 1, wherein a block of memory is a page of main memory.

10. (Currently amended) A dirty memory for a computer system, the dirty memory being operable to store dirty indicators, each dirty indicator being settable to a given value indicative that a block of memory associated therewith has been dirtied, wherein said dirty indicators being stored in groups with each group having associated therewith a validity indicator computed from the dirty indicator values of the group, wherein the dirty memory comprising comprises control logic operable on reading [[a]] said group to compute the validity indicator based on the dirty indicator values for the group to determine the an integrity of the group, wherein the control logic is also operable to reset each dirty indicator of a group and the validity indicator for the group after reading the group.

11. (Previously presented) The dirty memory of claim 10 configured to re-compute the validity indicator for a group each time a dirty indicator in the group is changed.

12. (Original) The dirty memory of claim 10, where the validity indicator is a parity indicator.

13.   (Original) The dirty memory of claim 10, wherein a group of dirty indicators plus the validity indicator occupy one memory word.

14.   (Original) The dirty memory of claim 10, wherein each dirty indicator comprises a single bit.

15.   (Original) The dirty memory of claims 10, wherein a validity indicator comprises a single bit.

16.   (Original) The dirty memory of claim 10, wherein a block of memory is a page of main memory.

17.   (Currently amended) A computer system comprising:

       a dirty memory and at least one processing set that comprises main memory, and

       a the dirty memory being operable to store dirty indicators, each dirty indicator being settable to a given value indicative that a block of memory associated therewith has been dirtied, wherein said dirty indicators being stored in groups with each group having associated therewith a validity indicator computed from the dirty indicator values of the group, wherein the dirty memory comprising comprises:

       control logic operable on reading [[a]] said group to compute the a validity indicator based on the dirty indicator values for the group to determine the an integrity of the group, wherein the control logic being also operable to reset each dirty indicator of a group and the validity indicator for the group after reading the group.

18.   (Currently amended) The computer system of claim 17, comprising a plurality of processing sets, that each comprising comprise main memory.

19.   (Currently amended) The computer system of claim [[17]] 18, wherein the plurality of processing sets are operable in lockstep, the computer system comprising logic operable

4

to attempt to reinstate an equivalent memory state in the main memory of each of the plurality of processing sets ~~processor~~ following a lockstep error.

20. (Currently amended) A method of managing reinstatement of an equivalent memory state in ~~the~~ a main memory of a plurality of processing sets of a fault tolerant computer following a lock step error, the method comprising ~~the~~ a performance of at least one cycle of copying any block of memory that has been dirtied from a first processing set to each other processing set, each cycle comprising:

_____interrogating a dirty memory comprising dirty indicators settable to indicate dirtied blocks of memory, said dirty indicators being stored in groups with each group having associated therewith a validity indicator computed from the dirty indicator values of the group, said interrogation comprising computing a validity indicator value based on the dirty indicator values for the group to determine ~~the~~ an integrity of the group; and

_____identifying all dirty indicators of a group as representing a dirtied state where a validity indicator value based on the dirty indicator values read for the group is computed that is different from a validity indicator value read for that group.

21. (Original) The method of claim 20, where the validity indicator is a parity indicator.

22. (Cancelled)

23. (Currently amended) The method of claim [[22]] 20, further comprising copying the block of memory associated with each dirty indicator of a group from the memory to another memory where a validity indicator value based on the dirty indicator values read for the group is computed that is different from a validity indicator value read for that group.

24. (Cancelled)

25. (Original) The method of claims 20, wherein a group of dirty indicators plus the validity indicator occupy one memory word.

26. (Original) The method of claim 20, wherein each dirty indicator comprises a single bit.

27. (Original) The method of claim 20, wherein a validity indicator comprises a single bit.

28. (Original) The method of claim 20, wherein a block of memory is a page of main memory.

29. (Previously presented) The method of claim 20, comprising re-computing the validity indicator for a group each time a dirty indicator in the group is changed.

30. (Currently amended) A dirty memory control logic for a computer system, the dirty memory being operable to store dirty indicators, each dirty indicator being settable to a given value indicative that a block of memory associated therewith has been dirtied, wherein said dirty indicators being stored in groups with each group having associated therewith a validity indicator computed from the dirty indicator values of the group, wherein the dirty memory comprises the control logic being operable on reading [[a]] said group to compute a validity indicator value based on the dirty indicator values for the group to determine the an integrity of the group, wherein the control logic is configured to identify identifies all dirty indicators of a group as representing a dirtied state where it computes a validity indicator value based on the dirty indicator values read for the group that is different from a validity indicator value read for that group.

31. (Currently amended) The dirty memory control logic of claim 30, wherein if operable to cause the block of memory associated with each dirty indicator of a group to be copied from the memory to another memory where the control logic computes a validity indicator value based on the dirty indicator values read for the group that is different from a validity indicator value read for that group, the control logic is operable to cause the

6

block of memory associated with each dirty indicator of the group to be copied from the memory to another memory.

32.   (Currently amended) A dirty memory control logic for a computer system, the dirty memory being operable to store dirty indicators, each dirty indicator being settable to a given value indicative that a block of memory associated therewith has been dirtied, wherein said dirty indicators being stored in groups with each group having associated therewith a validity indicator computed from the dirty indicator values of the group, wherein the control logic being operable on reading [[a]] said group to compute a validity indicator value based on the dirty indicator values for the group to determine ~~the~~ an integrity of the group, wherein the control logic being also operable to reset each dirty indicator of a group and the validity indicator for the group after reading the group.

33.   (Currently amended) A ~~method of managing reinstatement of an equivalent memory state in the main memory of a plurality of processing sets of a fault tolerant computer following a lock step error, the method~~ computer system comprising ~~the performance of at least one cycle of copying any block of memory that has been dirtied from a first processing set to each other processing set, each cycle comprising~~:

    at least one processing set that comprises main memory; and

    a dirty memory operable to store dirty indicators, each dirty indicator being settable to a given value indicative that a block of memory associated therewith has been dirtied, wherein ~~interrogating a dirty memory comprising dirty indicators settable to indicate dirtied blocks of memory,~~ said dirty indicators being stored in groups with each group having associated therewith a validity indicator computed from the dirty indicator values of the group, ~~said interrogation comprising computing a validity indicator value based on the dirty indicator values for the group to determine the integrity of the group, and~~ wherein the dirty memory comprises:

        control logic operable on reading said group to compute a validity indicator based on the dirty indicator values for the group to determine an

7

integrity of the group, wherein the control logic is configured to identify ~~identifying~~ all dirty indicators of a group as representing a dirtied state where <u>it</u> <u>computes</u> a validity indicator value based on the dirty indicator values read for the group ~~is computed~~ that is different from a validity indicator value read for that group.

34.   (Currently amended) The ~~method~~ <u>computer system</u> of claim 33, ~~further comprising copying the block of memory associated with each dirty indicator of a group from the memory to another memory where~~ <u>wherein if the control logic computes</u> a validity indicator value based on the dirty indicator values read for the group ~~is computed~~ that is different from a validity indicator value read for that group<u>, the control logic is operable to cause the block of memory associated with each dirty indicator of the group to be copied from the memory to another memory</u>.

35.   (Currently amended) A method of managing reinstatement of an equivalent memory state in ~~the~~ <u>a</u> main memory of a plurality of processing sets of a fault tolerant computer following a lock step error, the method comprising ~~the~~ <u>a</u> performance of at least one cycle of copying any block of memory that has been dirtied from a first processing set to each other processing set, each cycle comprising:

interrogating a dirty memory comprising dirty indicators settable to indicate dirtied blocks of memory, said dirty indicators being stored in groups with each group having associated therewith a validity indicator computed from the dirty indicator values of the group, said interrogation comprising computing a validity indicator value based on the dirty indicator values for the group to determine ~~the~~ <u>an</u> integrity of the group, and

resetting each dirty indicator of a group and the validity indicator for the group after reading the group.

8